

## Задача А. Декомпозиция

Имя входного файла: `decomposition.in`  
Имя выходного файла: `decomposition.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Рассмотрим дерево  $T$ . Назовем деревом декомпозиции корневое дерево  $D(T)$ .

Выберем любую из вершин дерева  $T$ , назовем ее  $r$ . Рассмотрим все компоненты связности дерева  $T$ , после удаления вершины  $r$ :  $S_1, S_2, \dots, S_k$ . Тогда корнем  $D(T)$  будет вершина  $r$ , а детьми  $r$  в  $D(T)$  будут  $D(S_1), D(S_2), \dots, D(S_k)$ .

Вам задано  $T$ . Найдите дерево декомпозиции, высота которого не более 20. Высотой дерева называется максимальное число вершин, которые может содержать простой путь начинающийся в корне.

### Формат входных данных

Первая строка содержит  $n$  — число вершин дерева  $T$  ( $1 \leq n \leq 2 \cdot 10^5$ ).

Следующие  $n - 1$  строк содержат ребра дерева. Каждое ребро описывается парой чисел  $v_i, u_i$  — концы ребра ( $1 \leq v_i, u_i \leq n$ ).

### Формат выходных данных

Выведите  $n$  чисел:  $i$ -е число — родитель вершины  $i$  в дереве декомпозиции, если вершина является корнем, выведите 0.

### Примеры

decomposition.in	decomposition.out
3 1 2 2 3	2 0 2
9 3 2 4 2 1 2 5 1 1 6 7 6 6 8 8 9	0 1 2 2 1 1 6 6 8

## Задача В. Центроиды дерева

Имя входного файла: `centroid.in`  
Имя выходного файла: `centroid.out`  
Ограничение по времени: 5 секунд  
Ограничение по памяти: 256 мегабайт

Дано дерево из  $n$  вершин. У каждой вершины есть цвет. Нужно обработать  $q$  запросов  $(v_i, c_i)$ : найти расстояние от  $v_i$  до ближайшей к  $v_i$  вершины цвета  $c_i$ . Расстоянием между вершинами называется минимальное количество рёбер в пути между ними.

### Формат входных данных

На первой строке число  $n$  ( $1 \leq n \leq 10^5$ ), следующая строка содержит числа  $p_1, p_2, \dots, p_{n-1}$ .  $0 \leq p_i < i$ .  $p_i$  – отец вершины  $i$  в дереве. Далее строка с числами  $a_0, a_1, \dots, a_{n-1}$ .  $0 \leq a_i < n$ .  $a_i$  – цвет вершины  $i$ . Далее строка с числом  $q$  ( $1 \leq q \leq 10^5$ ). Следующие  $q$  строк содержат запросы  $v_i q_i$  ( $0 \leq v_i < n$ ,  $0 \leq c_i < n$ ).

### Формат выходных данных

Для каждого запроса выведите одно число – расстояние до ближайшей вершины нужного цвета, или  $-1$ , если в дереве нет вершин такого цвета.

### Примеры

centroid.in	centroid.out
5	0 1 2 -1 2 1 2 1 1
0 1 1 3	
1 2 3 2 1	
9	
0 1	
0 2	
0 3	
1 0	
2 1	
2 2	
3 3	
3 1	
4 2	

## Задача С. Дорешивание

Имя входного файла: `upsolving.in`  
Имя выходного файла: `upsolving.out`  
Ограничение по времени: 5 секунд  
Ограничение по памяти: 256 мегабайт

В Летней Компьютерной Школе есть  $n$  параллелей, каждая из которых живёт в своём домике. Все параллели пронумерованы от 1 до  $n$  от младших к старшим. Периодически школьник, дорешивающий прошедшие практики у себя в домике, не справляется с задачей и идёт за помощью к товарищам из более старшей параллели.

Некоторые пары домиков соединены тропинками, всего есть  $n - 1$  такая тропинка. Все тропинки имеют одинаковую длину, по тропинке можно ходить между двумя домами, которые она соединяет, и только между ними. От любого домика можно дойти до любого другого домика, используя только данные тропинки.

Если у школьника из параллели  $k$  не получается решить задачу, он из своего домика с номером  $k$  идёт просить помощи до какого-нибудь домика с номером, большим  $k$ . Поскольку ему не хочется тратить ни секунды драгоценного времени, он выбирает ближайший подходящий домик. Школьники из параллели  $n$  всегда решают свои задачи сами, так как им не к кому обратиться.

Вам дано описание тропинок между домами. Для каждого  $k$  от 1 до  $n - 1$  определите минимальное расстояние, которое школьник из параллели  $k$  пройдёт в случае проблем с решением задачи.

### Формат входных данных

Первая строка входных данных содержит целое число  $n$  ( $1 \leq n \leq 200\,000$ ) — количество параллелей в ЛКШ.

В  $i$ -й из следующих  $n - 1$  строк содержатся два целых числа  $a_i$  и  $b_i$  ( $1 \leq a_i, b_i \leq n$ ,  $a_i \neq b_i$ ) — номера домиков, которые соединяет  $i$ -я тропинка.

Гарантируется, что каждую пару домиков соединяет не более одной тропинки, и что из любого домика можно дойти до любого другого.

### Формат выходных данных

Выведите  $n - 1$  строку,  $i$ -я из них должна содержать целое число  $d_i$  — расстояние до ближайшего домика с номером, большим  $i$ , от домика параллели  $i$ .

### Примеры

upsolving.in	upsolving.out
5	1
1 4	1
5 2	2
3 1	3
1 2	
5	1
4 3	2
3 5	1
5 1	2
1 2	

## Задача D. Пути в дереве

Имя входного файла: `tree-paths.in`  
Имя выходного файла: `tree-paths.out`  
Ограничение по времени: 10 секунд  
Ограничение по памяти: 512 мегабайт

Дано дерево из  $n$  вершин. Найдите для каждого  $d$  от 1 до  $n - 1$  число путей длины  $d$ .

### Формат входных данных

Первая строка содержит  $n$  — число вершин дерева ( $1 \leq n \leq 50000$ ).

Следующие  $n - 1$  строк содержат ребра дерева. Каждое ребро описывается парой чисел  $v_i, u_i$  — концы ребра ( $1 \leq v_i, u_i \leq n$ ).

### Формат выходных данных

Выведите  $n - 1$  число:  $i$ -е число — число путей длины  $i$ .

### Примеры

tree-paths.in	tree-paths.out
3	2
1 2	1
2 3	
9	8
3 2	10
4 2	10
1 2	6
5 1	2
1 6	0
7 6	0
6 8	0
8 9	